ArcGIS Server: Optimizes Map Services and Caches for performance

Jeremiah Lindemann

ESRI

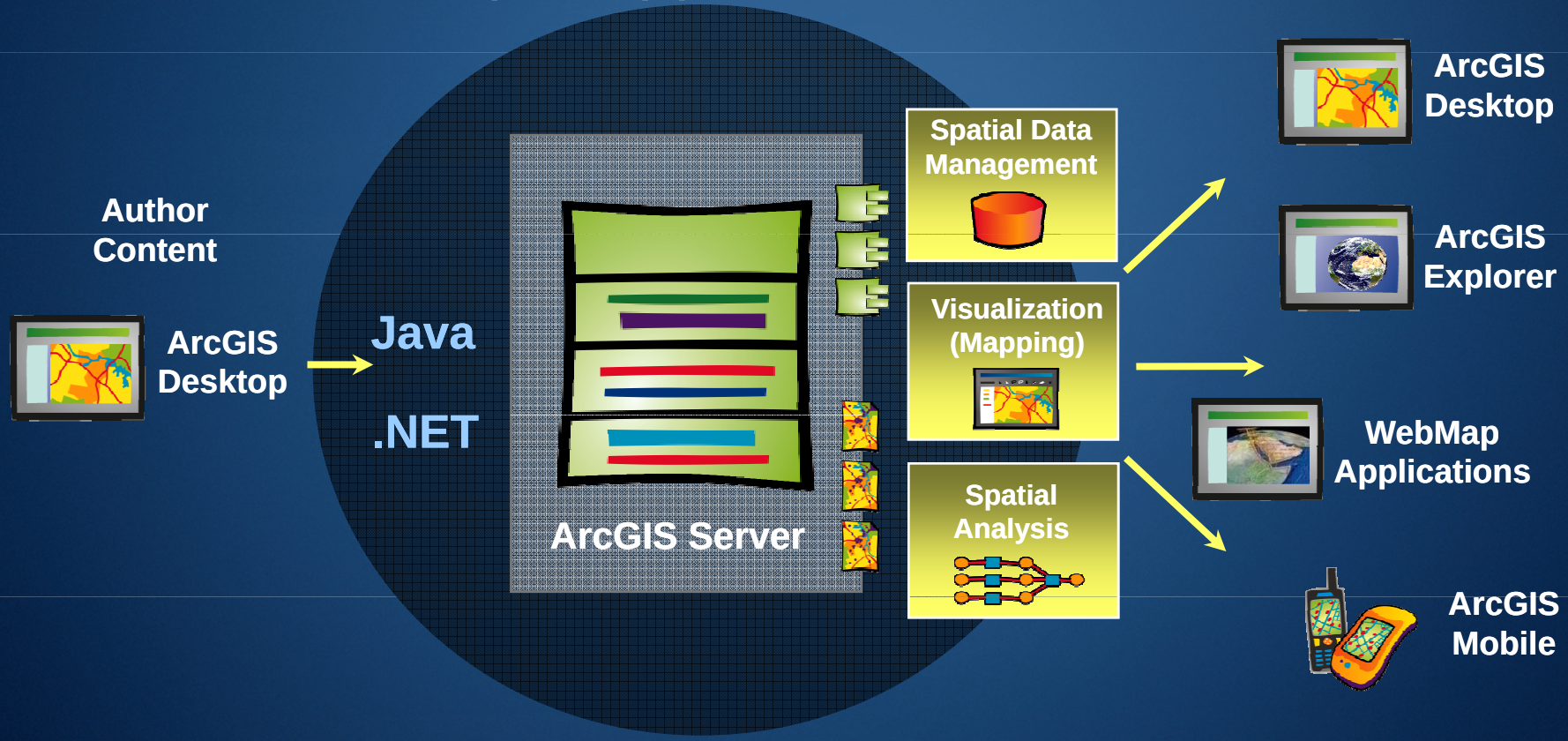
UGIC 2008

Agenda

- Optimizing your .mxd to publish as a map service
  - Examples of symbology, graphics and label choices that can lead to poor performance
- Building caches to boost performance
  - How to create a cache
  - Caching is an investment- discuss the many options to consider first

ArcGIS Server 9.3

- Complete & Integrated server-based GIS
- Out-of-the-box applications and services
- Rich developer opportunities



Performance recommendations

First tune the .mxd

- As a rule of thumb, if the .mxd loads slow the map service will be slow
- Is the sde data correctly tuned (indexes, compress, etc)?
- If using native raster data, are pyramids being used?
- Has the .mxd been saved many times? Perhaps save a new mxd for a smaller file size
  - http://support.esri.com/index.cfm?fa=knowledgebase.techarticles.articleShow&d=33187

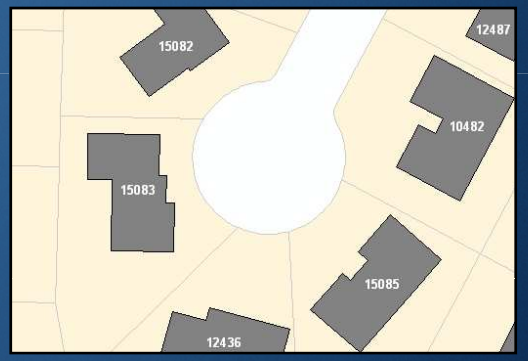
Picture yourself as the end user

- Is there too much data to cause confusion?
  - The total number of layers
  - The number of layers that can be turned on at once (scale dependencies can limit this)
- Would it be better to create separate web applications that are focused to specific datasets rather than one website with all the data?
- Will the user be able to distinguish all the tools available?

Optimizing map documents for publication

- Intelligently authored maps can improve performance
- Top tips for optimizing a map document:
  - Use scale thresholds for rendering of layers and labels
  - Remove any unnecessary layers or data frames
  - Use annotation instead of labels
  - Avoid complex symbology and label effects

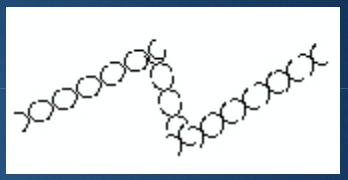
Unless you plan to cache!



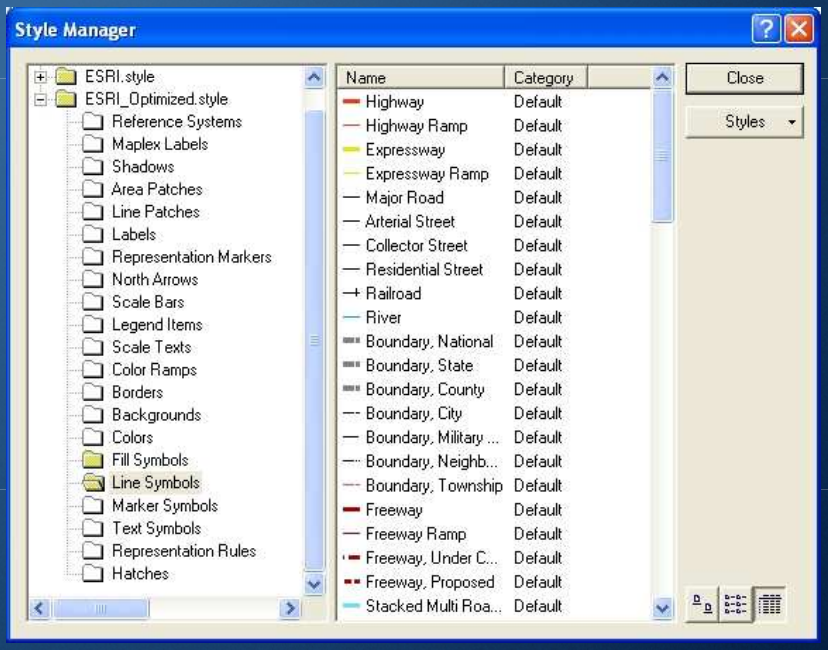
- Can increase speed up to 60%
- Optimization can also improve cache generation speed

Optimizing symbology (non-cached services)

- Avoid complex symbols
  - Multi-layer symbology
  - Outlines
  - Specialized fill patterns

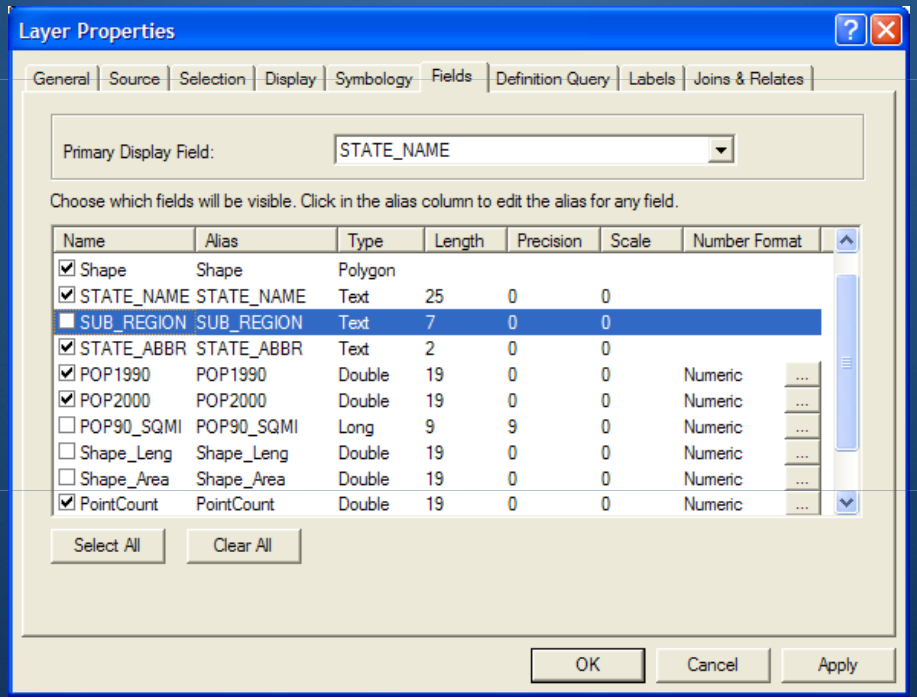


- Use the ESRI_Optimized style set
  - Simple line and fill symbols
  - Designed for faster drawing

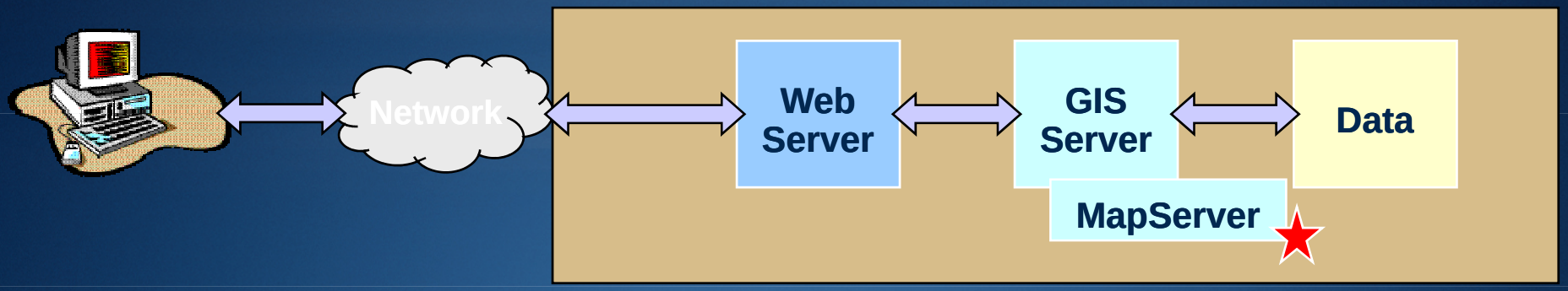


Optimizing map documents for publication

- Use definition queries
- Determine which layers are relevant on initial load
  - Try to minimize for background data
- Turn off field visibility for unnecessary fields
  - Queries will perform quicker



Performance in a non-cached service

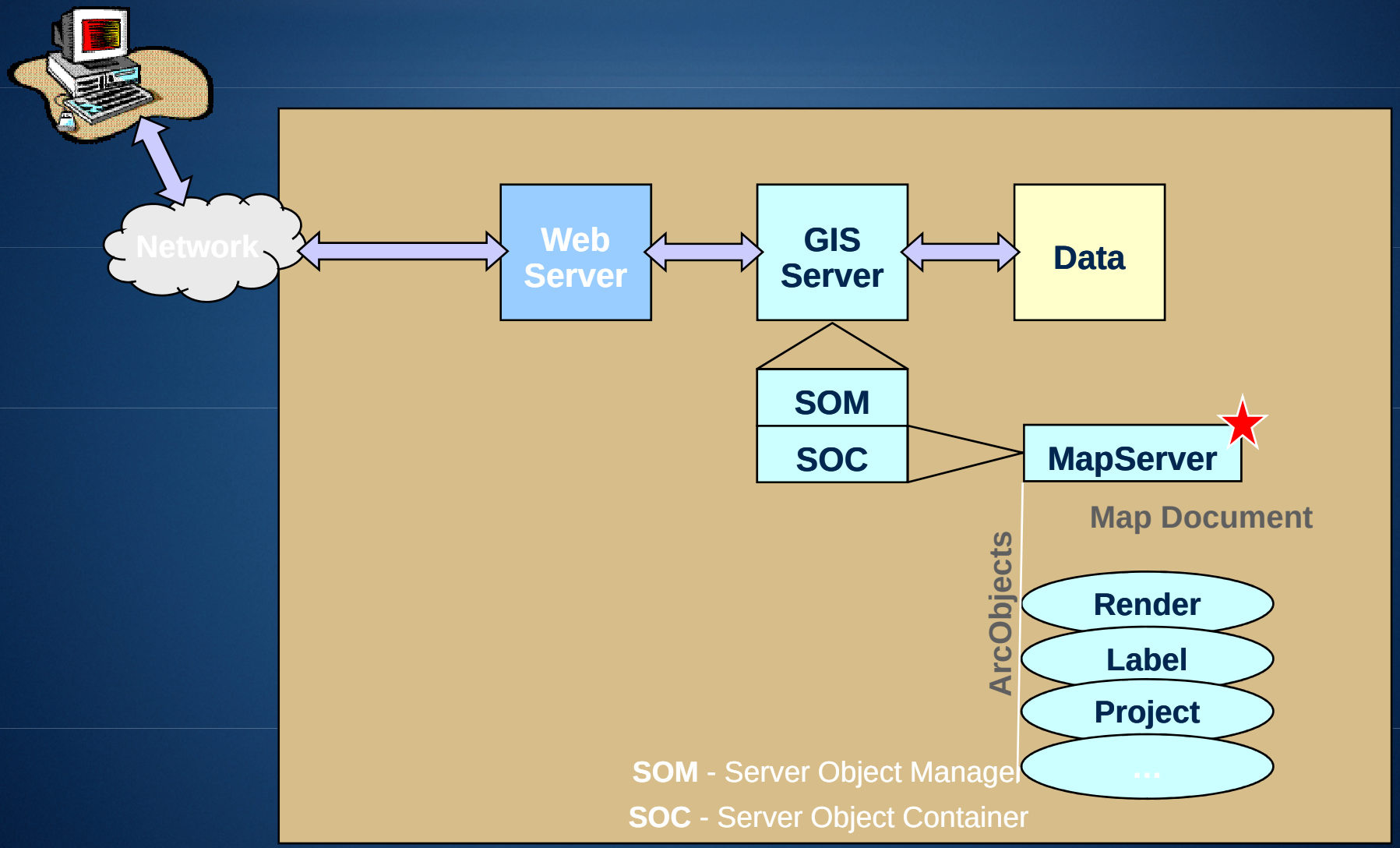


- Non-cached (dynamic) service communication flow
- Improving performance
  - Decrease MapServer tasks

Simplify the map document

- Map (scale dependency, data)
- Symbology (marker, line, polygon)
- Text

A closer look



Simplify the symbology

- General principles for all symbols
  - Points, lines, polygons, and text

Slower speed

- Cartographic symbols
- Multilayer symbols
- Labels
- Bitmap (BMP) symbols

Faster speed

- Simple symbols
- Optimized style
- Annotation
- Windows Enhanced Metafile (EMF) symbols

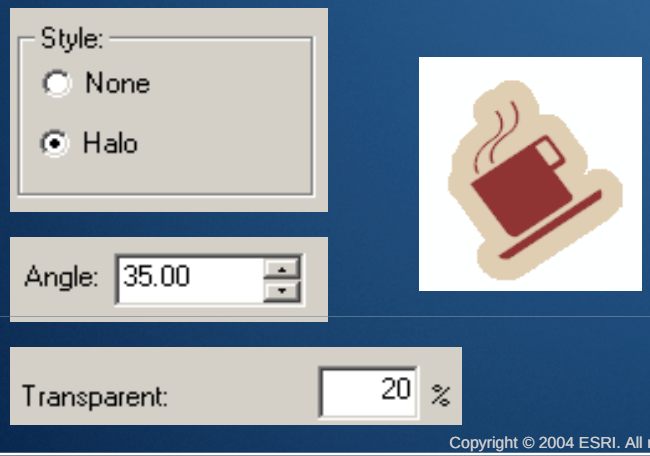
Marker symbology

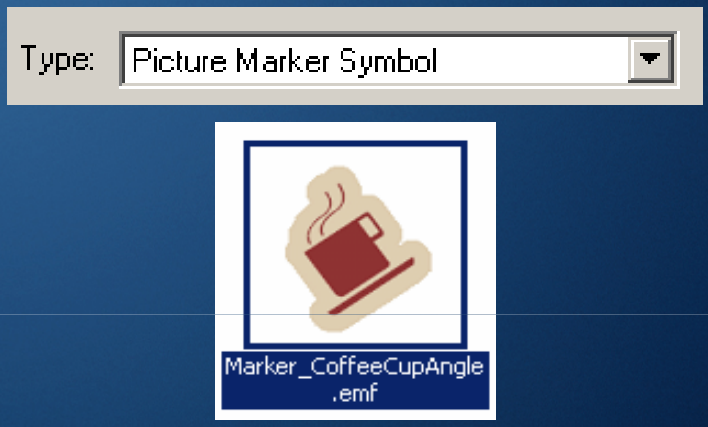
Slower speed

- Complex shapes or multiple layers
- Special effects
  - Halos
  - Transparency
  - Masking
  - Offsets
- Very large symbols

Faster speed

- Simple geometry
- Create an EMF using effects
  - Use picture marker symbol
  - No offset
- Use symbols < 60 pts





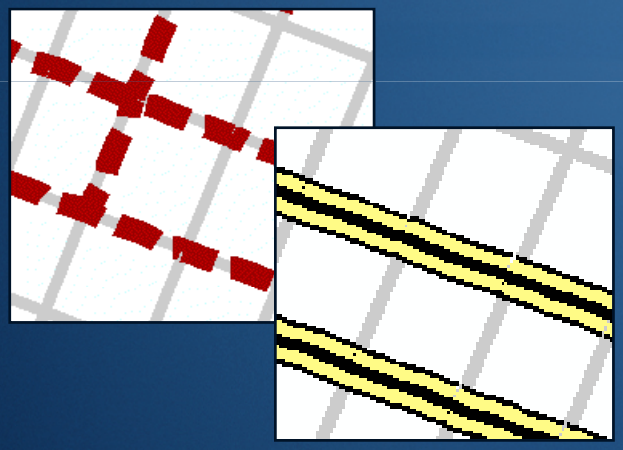
Line symbology

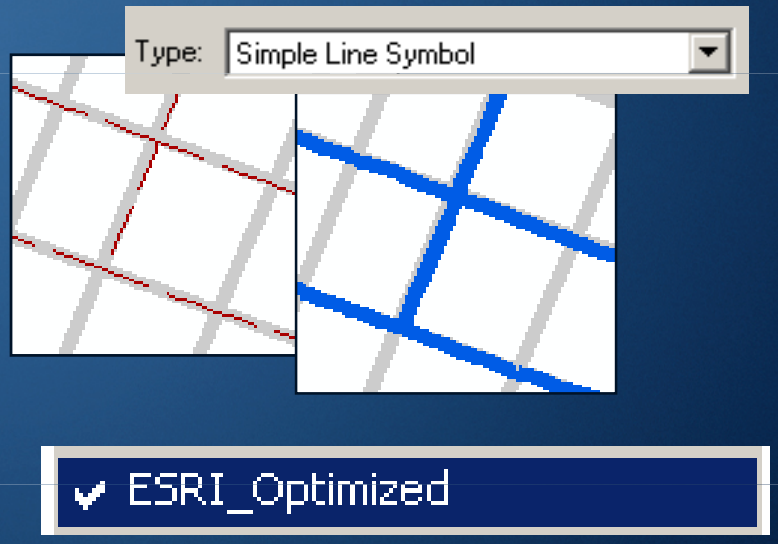
Slower speed

- Cartographic lines (e.g., dash)
  - Multilayer symbol

Faster speed

- Simple symbol
- ESRI_Optimized style





Line symbology

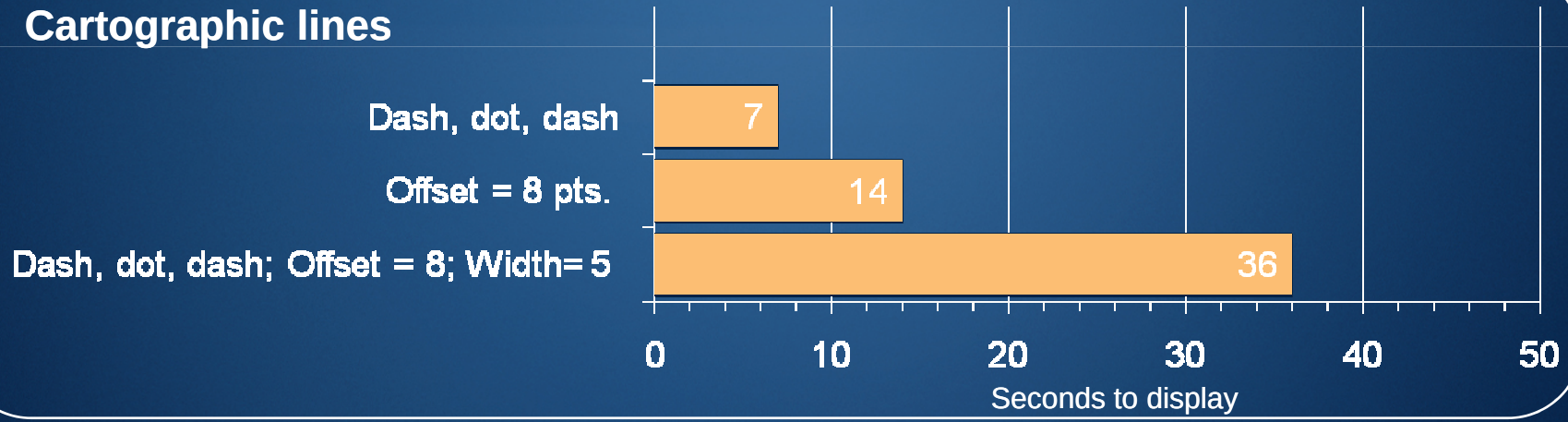
Slower speed

- Cartographic lines (e.g., dash)
  - Multilayer symbol
- Offsets, wide width

Faster speed

- Simple symbol
- ESRI_Optimized style
- No offset, width < 1.0 pt

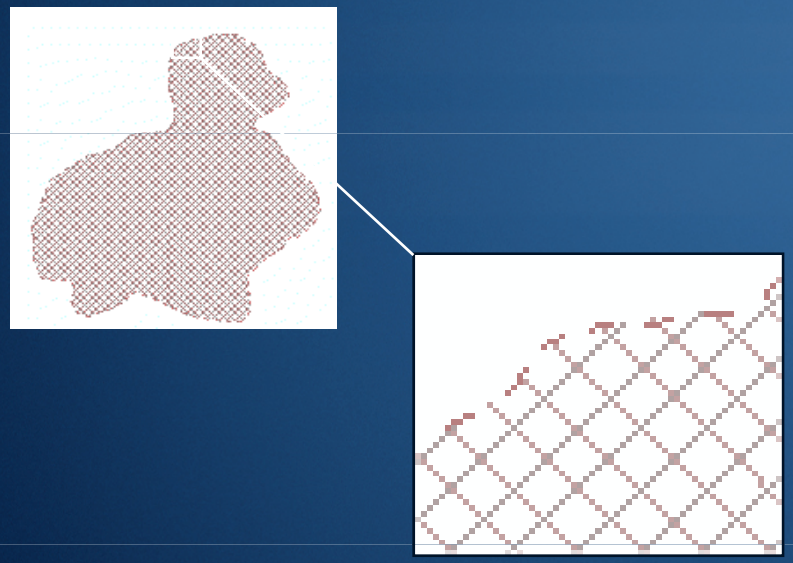
Cartographic lines



Fill symbology

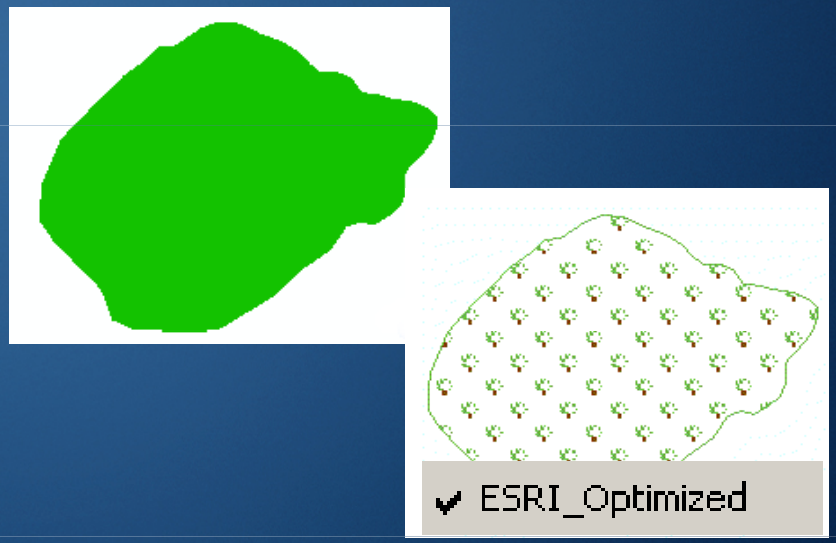
Slower speed

- Cartographic outlines
- Multilayer fill symbol



Faster speed

- Solid or no outline
- Simple fills, ESRI_Optimized style



Fill symbology

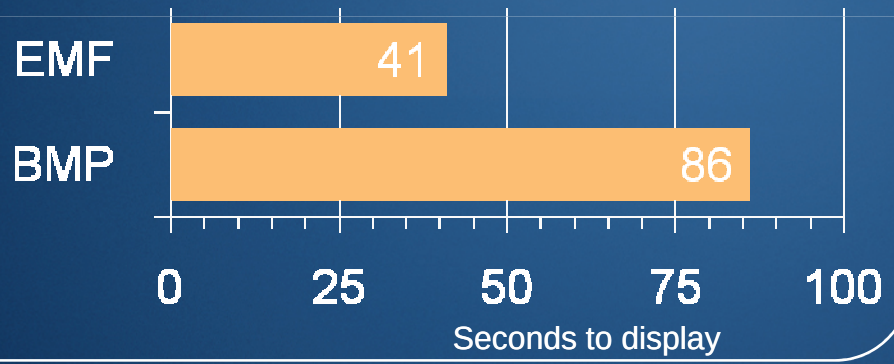
Slower speed

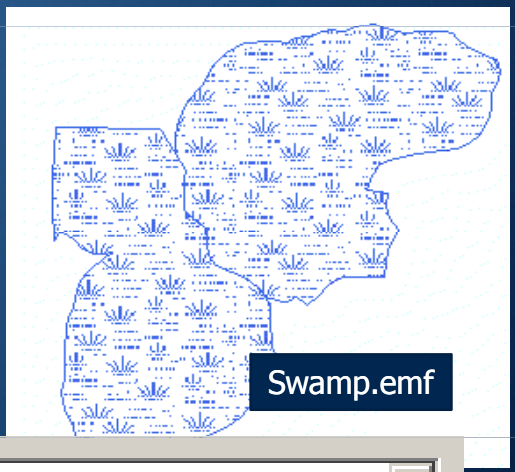
- Cartographic outlines
- Multilayer fill symbol
- BMP fills

Faster speed

- Solid or no outline
- Simple fills, ESRI_Optimized style
- Use EMF

Picture fills





Type: Picture Fill Symbol

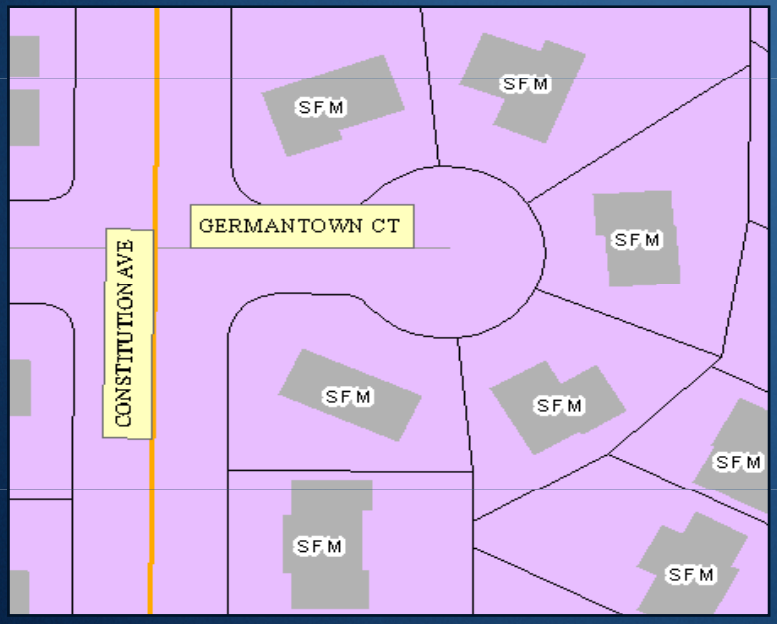
Text: Special effects

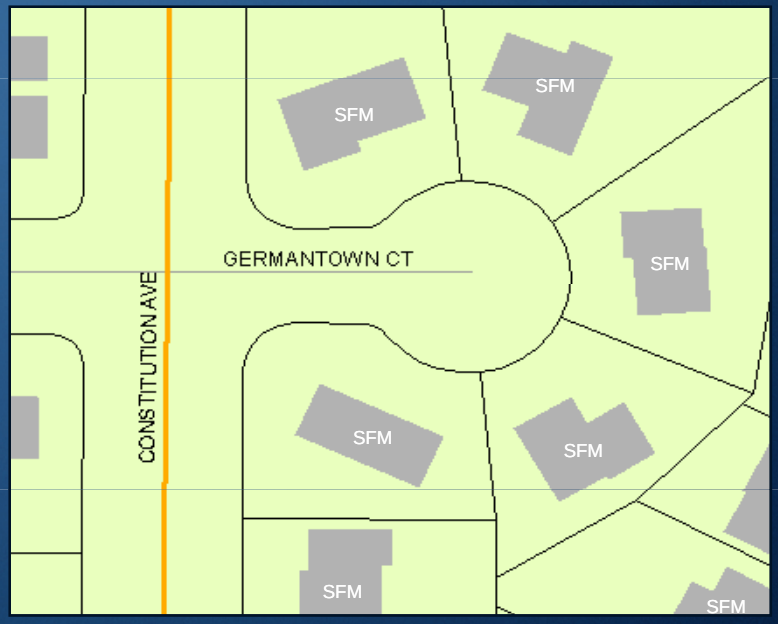
Slower speed

- Special effects (e.g., italics, halos, fill patterns)
- Text backgrounds, serif fonts (e.g., callout boxes)
- Very large text size (> 60 pts)

Faster speed

- Lighter background color or shadow effect
- Use wider line and character spacing
- Use appropriate font size (10 pts best)





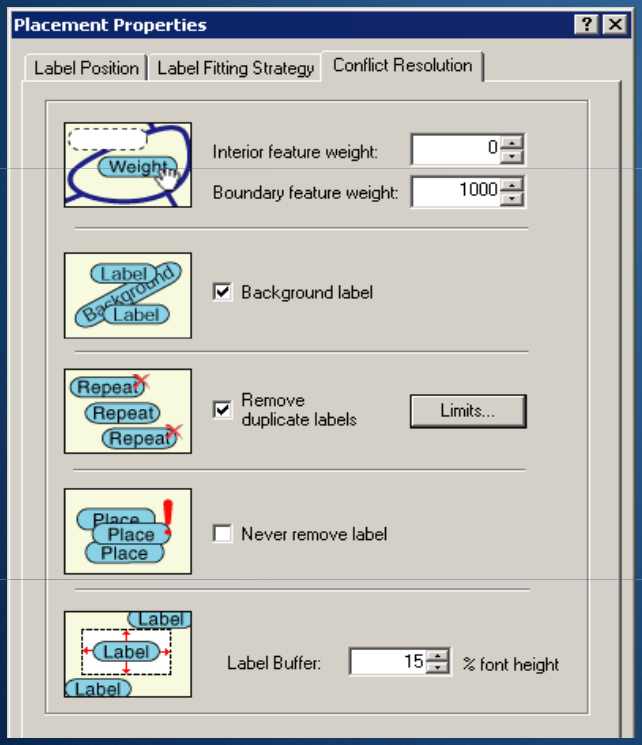
Text: Labels vs. annotation

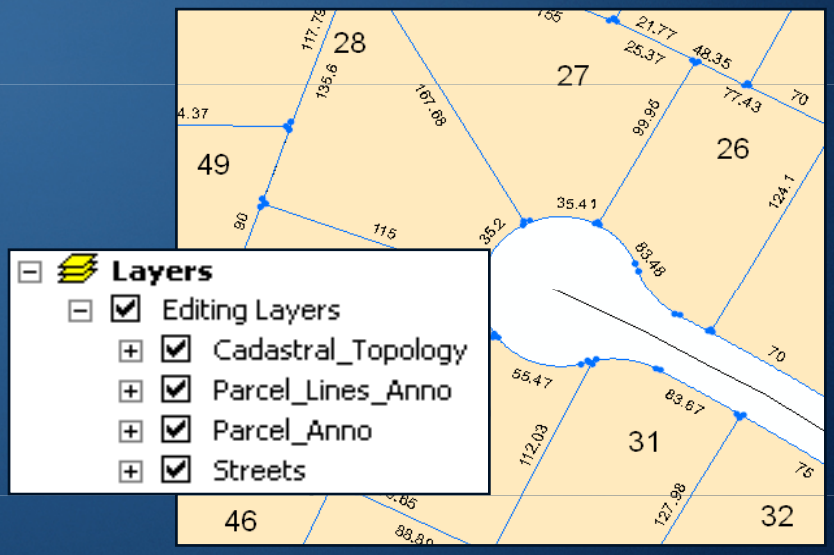
Slower speed

- Map annotation
- Complex labeling, Maplex
- Labels with complex SQL or VB scripts

Faster speed

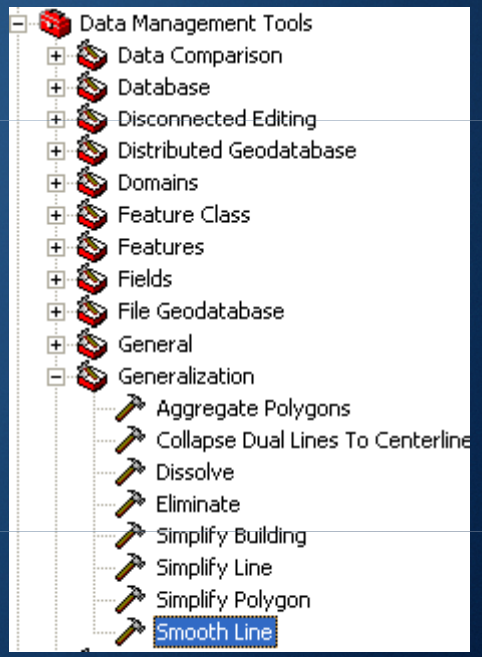
- Geodatabase annotation
- ESRI Standard Label Engine
- Create field with value for labels
- Use label properties sparingly





Use simplified geometry

- Use simplified representations of layers when displaying at smaller scales
- Ex: A detailed map of world coastlines may draw slowly at full scale. If this layer is simplified to have fewer vertices and line segments, it will draw much faster.
- Use scale dependencies and group layers to draw detailed layers only at larger (zoomed-in) scales.

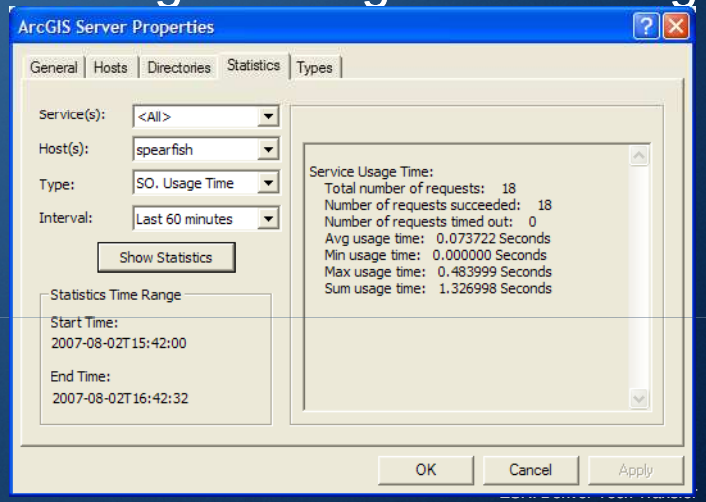


Optimizing map services summary

- If the .mxd performs slow in ArcMap, the map service will be slow
- Many things you can do to improve the performance of the map service
  - Individual changes may make only slight difference, but a cumulative change in tuning can have a drastic performance increase

Monitoring performance – other considerations

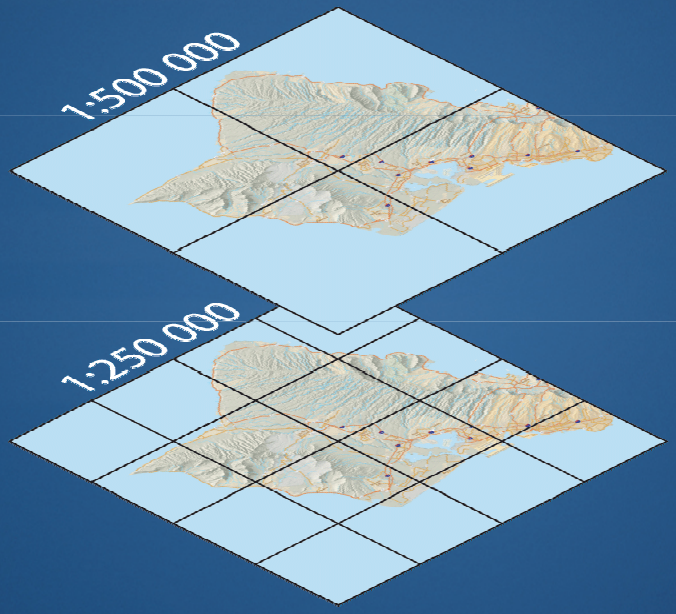
- How is the web server performing?
  - Do you have too many services running using CPU?
  - Are min/max instances set correctly?
  - Bandwith issues, etc
- Are pooled or non-pooled services running?
- Examine how many users are hitting your website
  - Monitor the statistics of each service through Manager or Catalog
    - If numbers are high, perhaps more SOC machines are needed



Building caches to boost performance

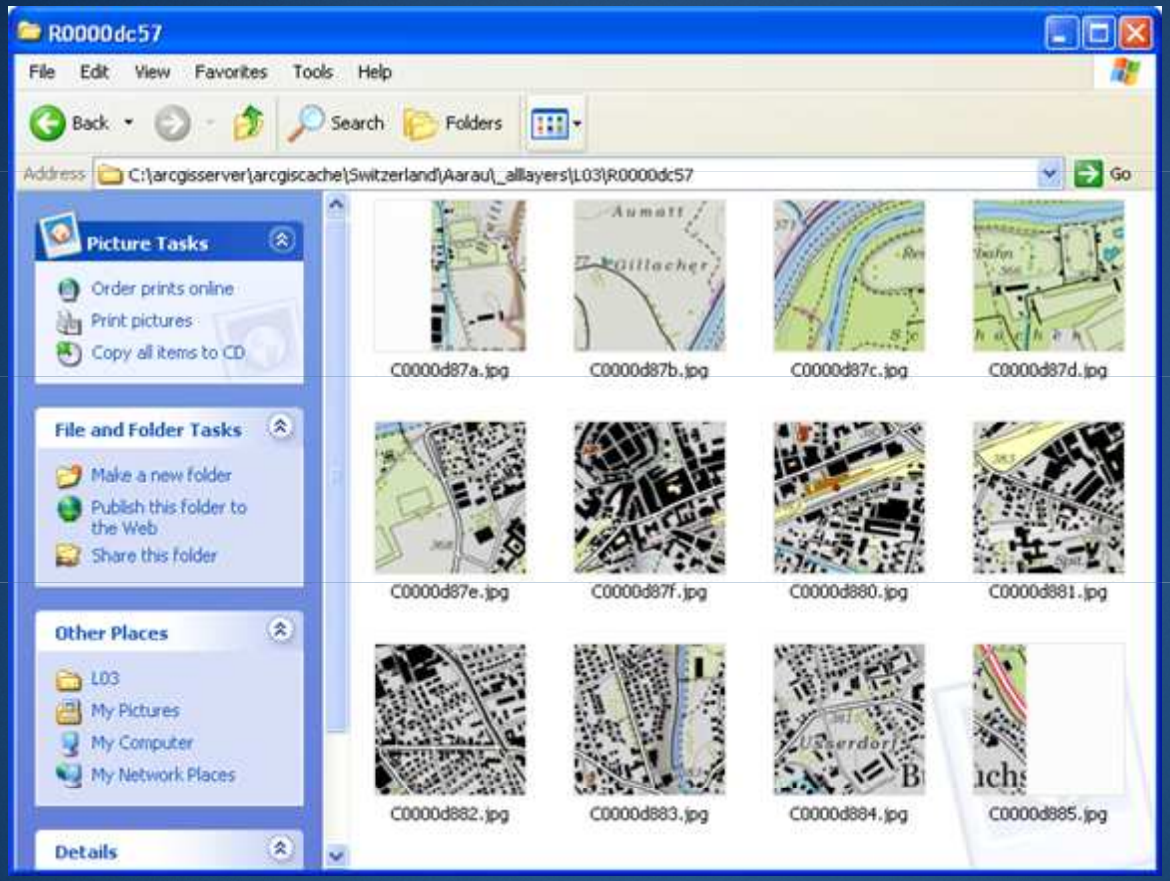
What is a map cache?

- A map cache is a set of map images that have been pre-rendered for rapid display.



- You create map caches at pre-determined scale levels.

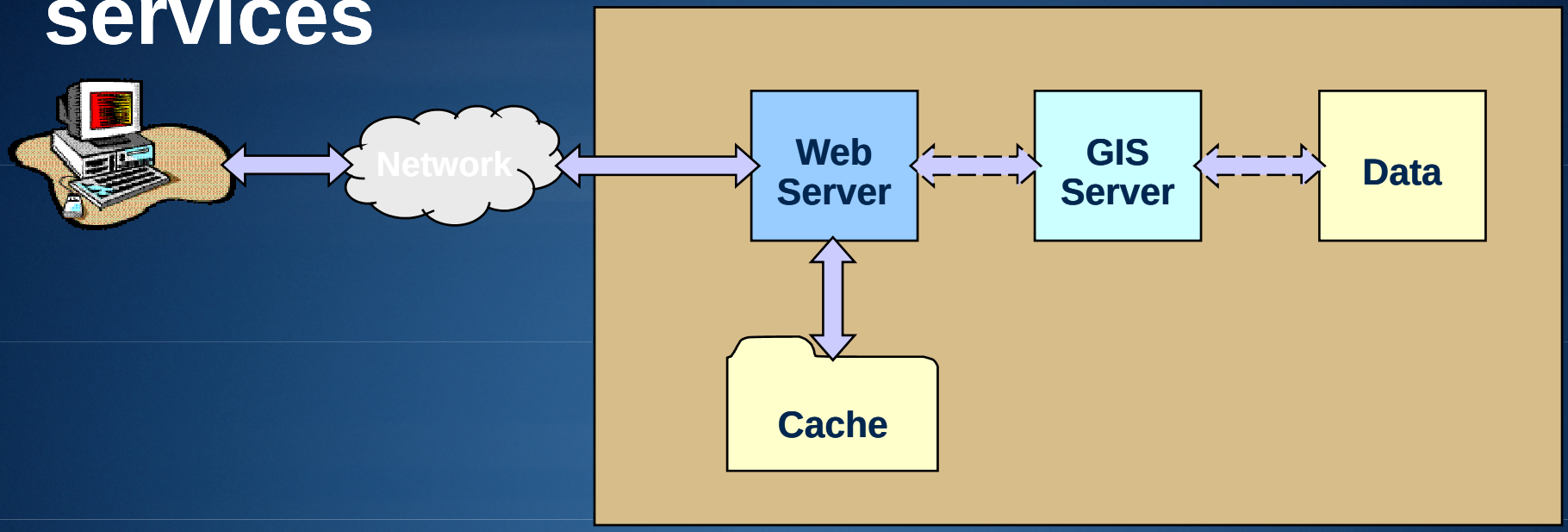
The cached images are stored on disk



Why cache a map?

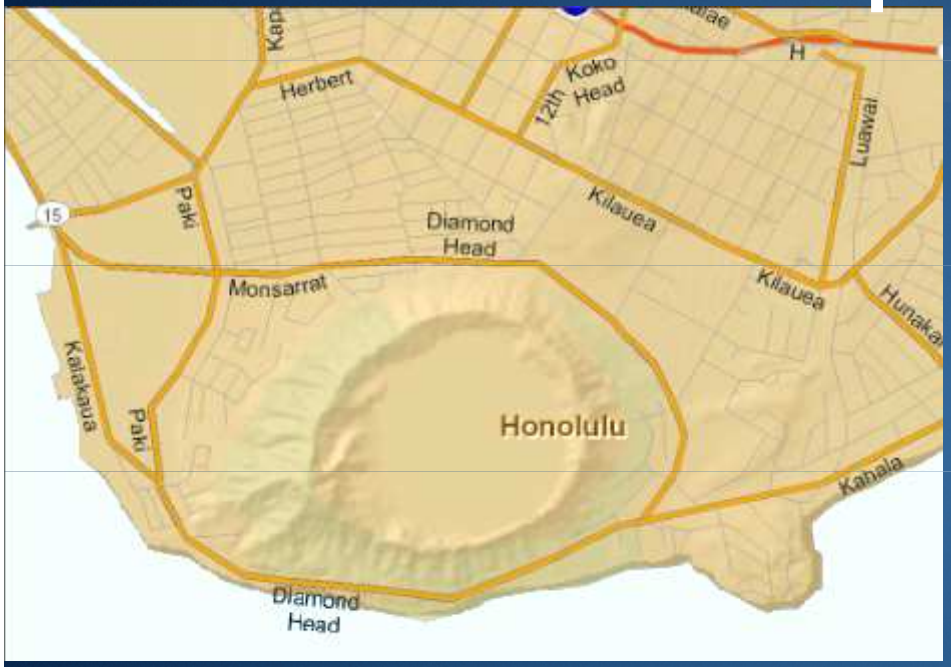
- Which is faster?
  - Let the server draw the map OR
  - Get the map image from a cache
- By caching, you only have to render the map once: When you create the cache.
- Caching is an *investment*.

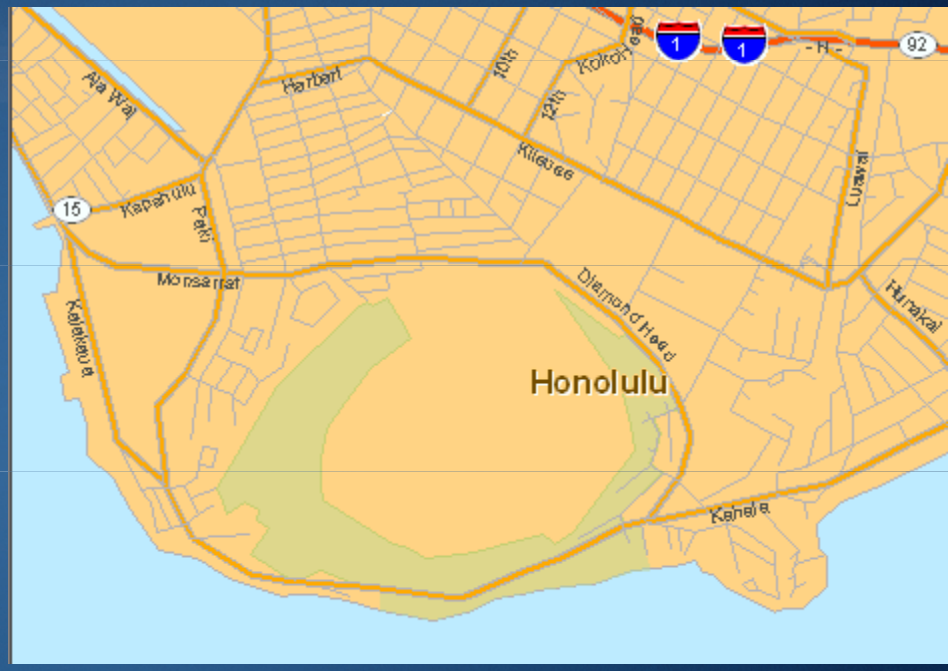
Communication flow for cached services



- Increased performance
  - Faster to retrieve cached image than have the server generate it
  - Render map once during cache generation
- Can still perform identify, query, and search of data

Keep Cartographic Quality and Map Performance with ArcGIS Server cached map services



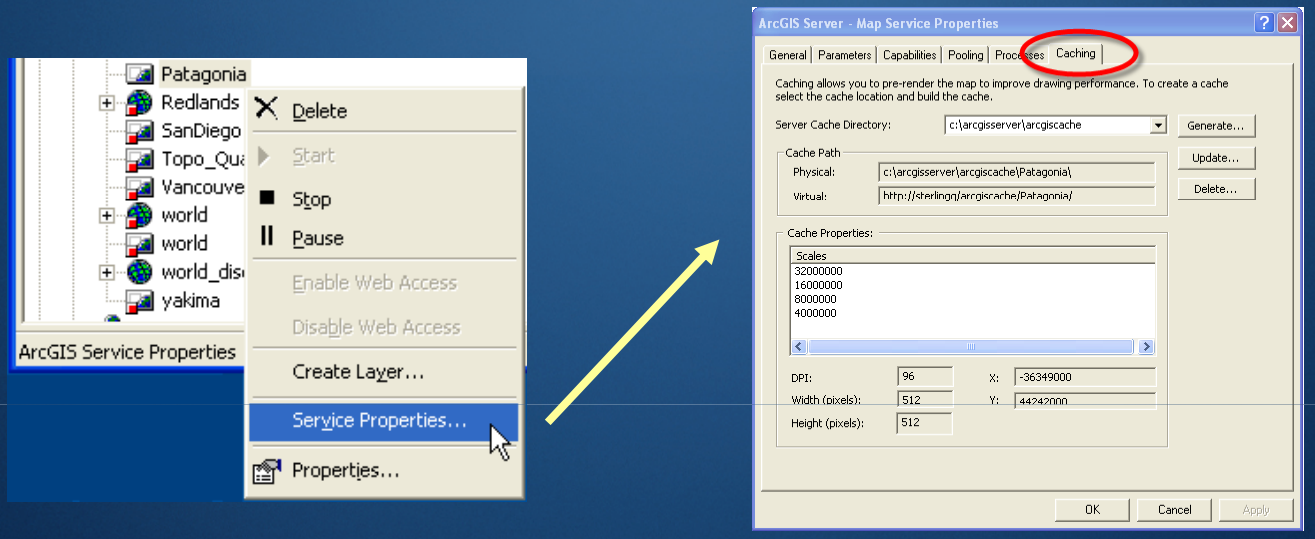


- Shaded Relief
- Transparent Layers
- Maplex Labeling

- Low-res relief
- Solid colors
- Annotation

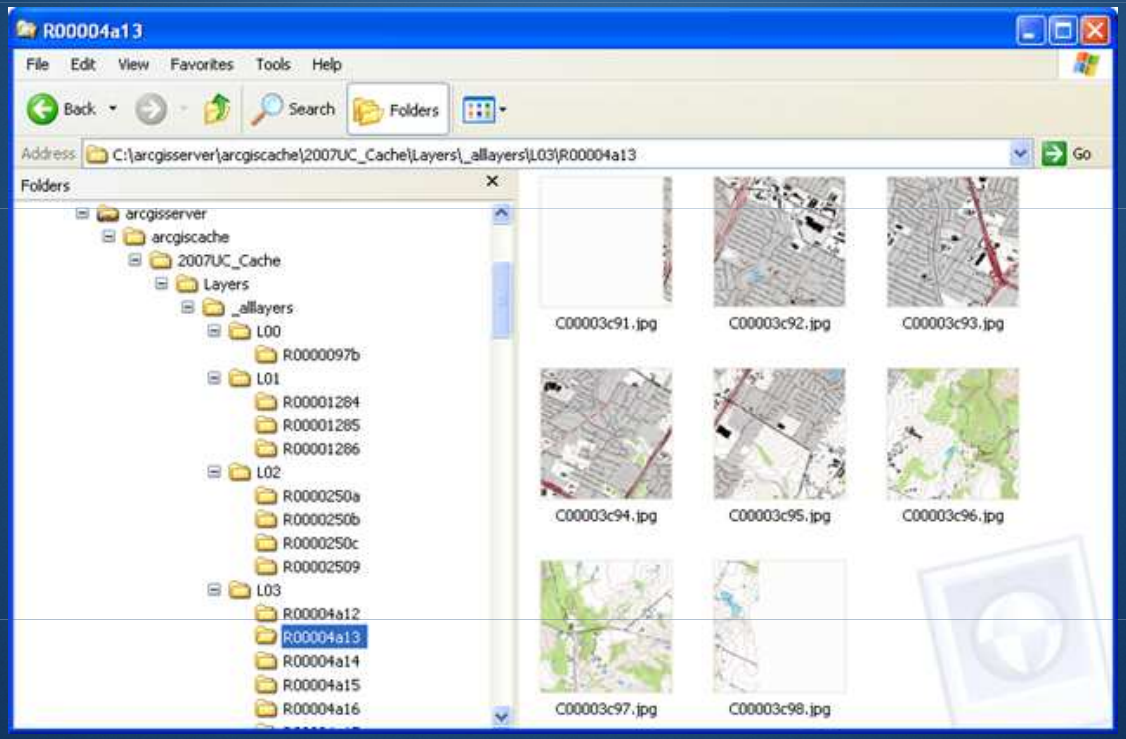
How do I create a map cache?

- You can create the cache in ArcCatalog
- You must have an existing map service running
- Use the Caching tab of the Service Properties



What happens during caching?

- The server draws the map at all of the scale levels you specified.
- Cached tiles are stored in a folder hierarchy in your server cache directory



Where is my server cache directory?

- If you installed all of the components of ArcGIS Server on one machine, you get a server cache directory by default.
- This default server cache directory is C:\arcgisserver\arcgiscache
- If you installed the components of ArcGIS Server over multiple machines, you must create a server cache directory in ArcCatalog or Manager

What types of maps should I cache?

- Base maps

- Maps that don't change frequently

- Maps you won't be editing

What type of cache should I create?

- Fused cache
  - Includes all layers in map in one “fused” image
  - Good performance
  - Can’t toggle layers on and off
- Multilayer cache
  - **Doesn’t work with out of the box ADF applications!**
    - An application built with Manager won’t read a multi-layer cache
    - ADF access via ExportMapImage method
  - Can choose groups of layers to be cached separately
  - Performance decreases with number of layer groups
  - Can toggle layers on and off

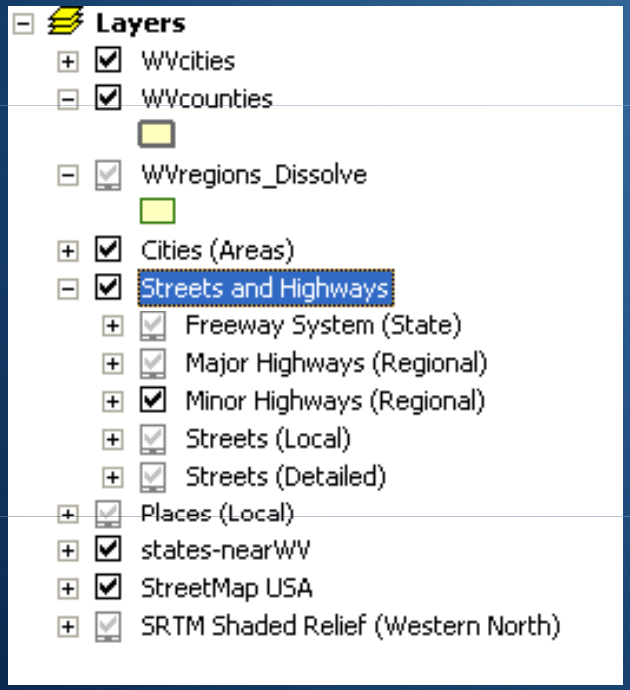
What type of cache should I create? (continued)

- Fused is recommended for most applications
- If you need to turn layers on and off, consider overlaying two fused map services instead of creating a multilayer cache
- ArcGIS Server 9.2 Service Pack 2 contains enhanced support for overlaying caches
- Multilayer caches work best in ArcMap

Cached map design tips

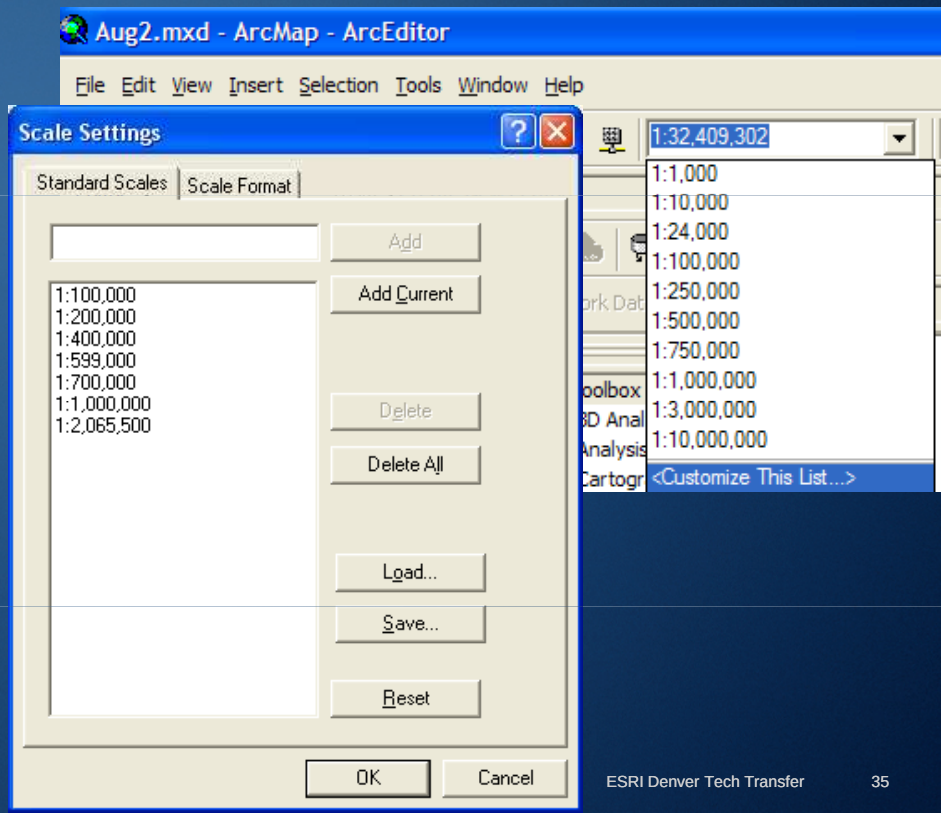
Layer Visibility

- Each Layer has its own scale properties
- Each layer is visible within an upper and lower scale threshold
- Layers are only visible at appropriate scale
- Together the layers work as one map



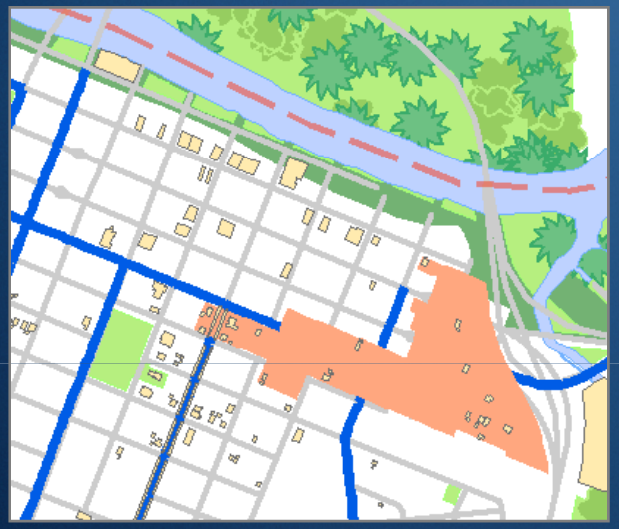
Map Scales (target scales)

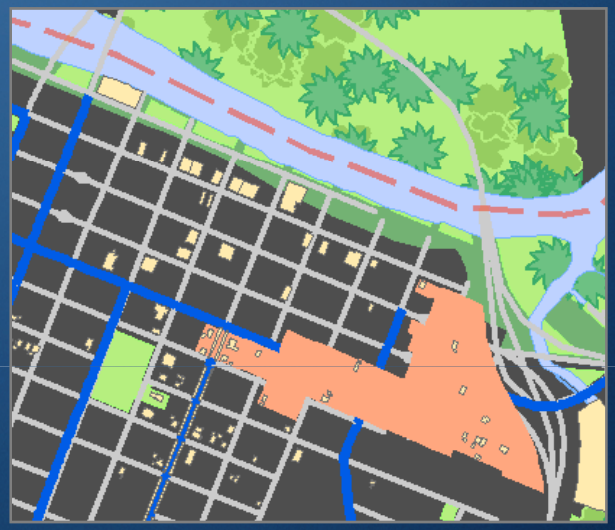
- You can define a list of scale thresholds for the entire map
- Test the map at specific scales as if you were to create caches at these scales



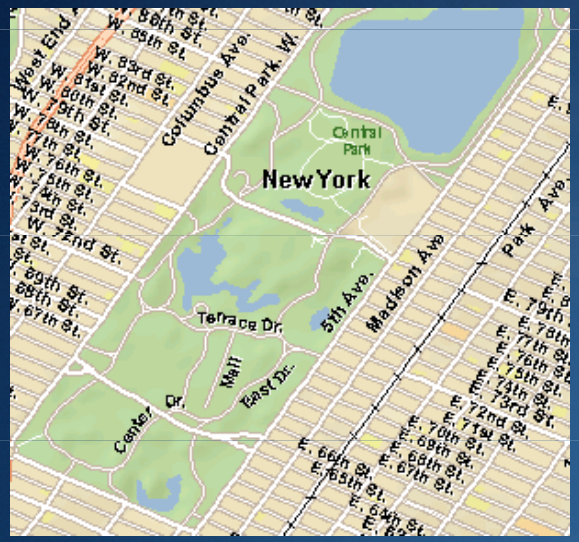
Background Color

- The background color is used to define the transparent part of the image
- Explicitly define the background color
  - If the background color is not defined, the transparent color will be set to 253,253,253 (nearly white)
- Use a color NOT used in the symbology





Anti-aliasing



- Smoothes the edges of labels and lines by blending them with the background.
- The resulting screen display quality can be better than standard rendering in ArcMap.
- Tiles are rendered at finer resolution by down sampling (takes twice as long to cache an area when using anti-aliasing)
- http://serverx.esri.com/antialiasingexamples/

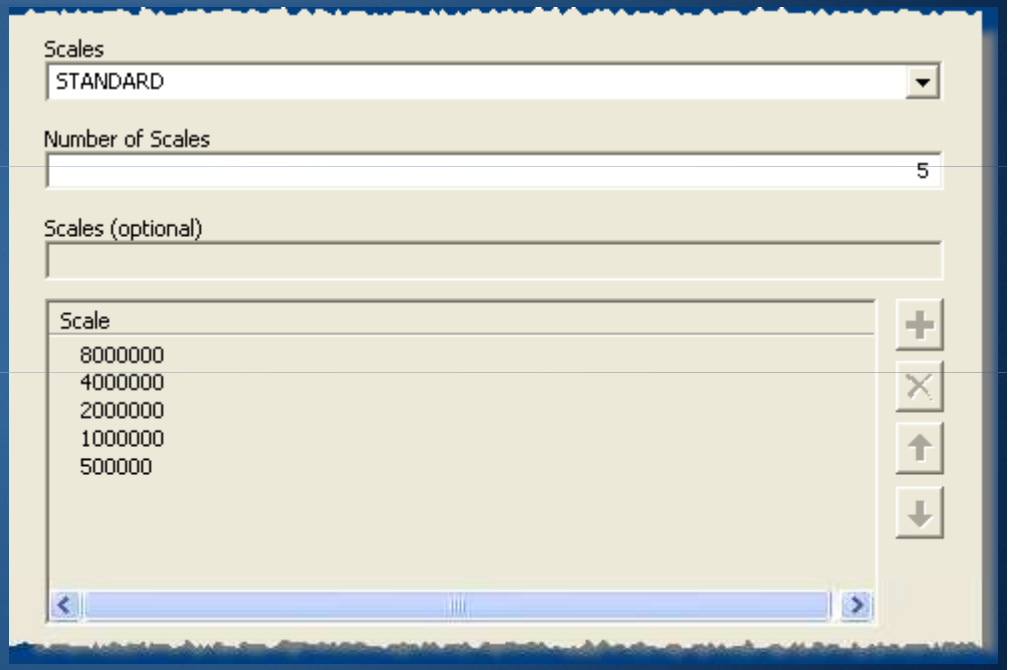
DPI (Dots per inch)

- Used to compute the resolution of an individual pixel for the cache tile that the server generates.

- If printing map caches you may find it useful to increase the DPI beyond the default (96) to match the output device's DPI setting. This will increase the number of the files in the cache.

Scales (Levels of Detail)

- Each cache has a set of user-defined scale levels
- Choose scale levels carefully!
  - They determine cache creation time and storage size on disk
  - Users will be limited to these scales
- Scales appear as LODInfos in conf.xml (LOD = “Levels of detail”)



How big will my cache be?

- The more detail, the bigger disk space required
- Code available from EDN: http://arcscripts.esri.com/details.asp?dbid=15045

Cache Tile Format

- PNG8 Transparency is stored in the color index palette, excellent browser support
- PNG24 Transparency value is stored in the image header. Versions of Internet Explorer less than version 7 do not support this type of transparency
- PNG32 —Supports large color variations (16 million colors) and transparency.
- JPEG —Supports large color variations (16 million colors) but does not support transparency

Cache tile format best practices

- Use JPEG for raster-based base maps such as imagery
- Use JPEG or PNG for vector-based base maps such as street maps
- Use PNG8 for overlay services that need to be supported in all browsers

Generating the map cache

Generating the map cache

- Can be time consuming
- Tip: Create the cache for a small area before building the whole thing.
  - Examine appearance of symbology, labels, etc.
  - Test the performance of the cache in your preferred client application
  - Make necessary adjustments and build again

Caching jobs are defined by…

- Cache tiling scheme
- Full extent of the map service (derived from the data frame full extent in the source MXD)

- For example:
  - Tiling origin is -180, 90
  - Map service extent is the bounding box of Colorado
  - Tiles will only be generated within and around the bounding box of Colorado
    - The tile rows and columns will be referenced from the tile origin (-180, 90)

Factors that influence cache creation time

- Geographic extent
- Number and choice of scale levels
- Complexity of the map
  - If it takes a long time to draw in ArcMap, it will take a long time to cache.
- Anti-aliasing
  - Generally twice as long to generate
- Data source type
  - Enterprise geodatabase vs. local copies of data
- Server resources
  - For example, SOC machines and available service instances
- Network bandwidth between SOC machine and cache directory

Factors that influence cache creation time

- Tile size also affects cache creation time
  - Larger size produces fewer tiles
    - Less disk space (block size)
    - Faster creation
    - Easier to manage
  - Smaller size
    - Allows partial update of the display
  - Takes approximately 5X as long and takes up 1Gb more of space when creating a cache at 128x128 tile size versus 512x512 tile size with the same data (Hawaii)
  - In most cases, it's best to keep the default of 512x512

Impact of scale selection: StreetMap USA

- 48 states
- Cached on 6 dual-CPU servers

Scale	Files	Creation Time
1:500K	4K	2 min
…	…	…
1:64K	0.3M	2 hours
1:32K	1.1M	4.5 hours
1:16K	4.7M	37 hours

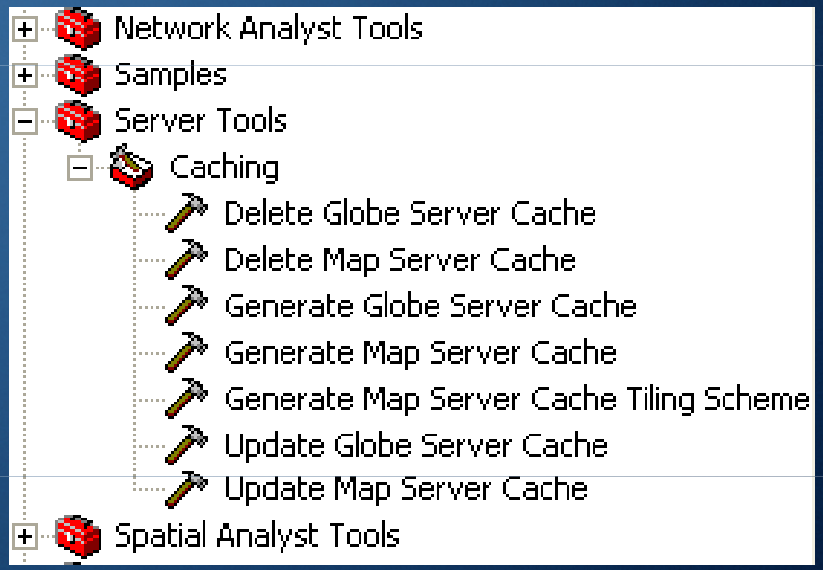
Total Size on Disk: 57 GB

Tips for large scope caching jobs

- When caching very large geographic areas break up caching job to distinct areas
  - Use UpdateMapServerCache at specific user defined extents in a script environment (Service Pack 2)
- Areas that don't need to be cached should be built using custom extents
  - Alaska, Hawaii, Continental US, but not all scale levels of the Pacific Ocean
  - You can cache the same map service using different full extents as long as your cache tiling schema doesn't change

Generating the map cache – 9.2 tools

- GenerateMapServerTilingScheme
  - Generates a tiling scheme that can be used to create caches for multiple services
- GenerateMapServerCache
  - Generates the cache for a map service
  - Works with either a pre-defined or a newly defined tiling scheme
  - Creates the cache and populates it
- UpdateMapServerCache
  - Updates the cache for a map service within a specified extent
  - Creates only missing and empty tiles OR Recreates all tiles
- DeleteMapServerCache
  - Deletes the cache for a map service



Updating a map cache

- Use Update Map Server Cache tool to update dynamic data
  - Don’t want to use for Real-Time/critical data but can use for semi-frequently updated data
  - Run as an automated Python script

http://blogs.esri.com/Dev/blogs/arcgisserver/archive/2008/01/04/Updating-map-caches-automatically_3A00_-The-key-to-caching-dynamic-data.aspx

Using map caches in client applications

ArcGIS Server Web mapping applications

- Latest ArcGIS 9.2 Service Pack *highly recommended*
  - SP2 allows for easier overlay of caches
- To get the performance benefit of the cache:
  - Navigation is limited to the cached scales
  - Cached tiles cannot be projected on the fly
- Differences between Java and .NET implementations
  - Check the ArcGIS Server Help online (http://webhelp.esri.com) to see specific instructions for your platform

Using map caches in 9.2 Service Pack 2

- Criteria to overlay multiple cached services in Java:
  - Projection must be the same
  - Scales that are common to both layers should match
- Criteria to overlay multiple cached services in .NET:
  - Projection must be the same
  - Scales that are common to both layers should match
  - Tile origin and tile size should be identical between services

What's ahead for caching/ArcGIS Server at 9.3

- Many internal enhancements for performance in Web ADF
- Ability to specify layers/fields to query/identify in Manager
- Easy workflow to define/update a cache
  - ArcGIS Manager and ArcGIS Desktop
- Support for on-demand map caching
  - Build your cache dynamically
- Use cache with new Javascript/REST APIs
  - Even faster presentation
  - Display map directly in website without Web ADF controls

Other useful tools

- Stress Server
  - Command line application to test a service to act as many users requesting map images
  - Used to help monitor image creation speed
  - Contact us for application
- Service Monitor
  - DOS command to tell inform you if services are up and running
  - http://arcscripts.esri.com/details.asp?dbid=15335

References & Useful sites

- Performance tips:
  - http://webhelp.esri.com/arcgisserver/9.2/dotNet/manager/publishing/map_service.htm
- Planning a map cache:
  - http://webhelp.esri.com/arcgisserver/9.2/dotNet/index.htm#manager/publishing/planning_a_map_cache.htm
- ArcGIS Server Development Blog:
  - http://blogs.esri.com/Dev/blogs/arcgisserver/